EFFECTIVE TODAY ~ ESSENTIAL TOMORROW

SIAMEASE TASKS

Simmonds and Associates
Copyright © 2001

TASKS AND THE MANAGEMENT THEREOF

Functional User Manual

SIAMEASE 3.0

SAP R/3 Automation

SIMMONDS AND ASSOCIATES

COPYRIGHT © 2001

TABLE OF CONTENTS

Chapter 1 FOREWORD 4
Chapter 2 DOCUMENT SPECIFICATIONS 4
Chapter 3 SETUP 5
IMPORTING TRANSPORT REQUESTS 5
REGISTRATION 5
SIAMEASE CONFIGURATION 6
TO CONFIGURE ADDITIONAL APPLICATIONS: 6
TO CONFIGURE ADDITIONAL PROJECT TYPES: 7
TO CONFIGURE ADDITIONAL PROJECTS 7
NAMING CONVENTIONS 7
Chapter 4 SIAMEASE TASK BUILDER ® 8
TASK DEFINITION 8
TO CREATE A NEW TASK 9
TO RECALL AN EXISTING TASK 9
RECORDING THE TRANSACTION 10
OTHER RECORDING CONSIDERATIONS 11
MANAGING THE PROBLEM 11
SIAMEASE LOGIC 11
RECORDING TIPS 12
ADVANCED RECORDING 12
RE-RECORDING A TASK 12
SELECTING PARAMETERS (INPUT FIELDS) 13
TO FLAG VARIABLE FIELDS 13
OTHER PARAMETER CONSIDERATIONS 15
FILLING MULTIPLE FIELDS WITH ONE VALUE 15
DATA FORMAT 15
DATA TEMPLATE 17
ADVANCED MAINTENANCE. 17
GENERATING THE TASK 18
OTHER GENERATION CONSIDERATIONS 18
TRANSPORTING 18
DATA FORMAT 18
DATA MAINTENANCE 18
INCLUDING CUSTOMER CODE 19
INCLUDE CURSOR POSITIONS 20
GLOBAL VALIDATIONS 20
MAINTAIN VARIANT FORMULAE 20
DATA SOURCE 21
SIMPLIFYING DATA CAPTURE 21
INTERFACING 22
Chapter 5 RUNNING A TASK 23
EXECUTING INDIVIDIAL SIAMEASE TASK 23
EXECUTE FROM THE SIAMEASE TASK BUILDER® 23
EXECUTE FROM THE SIAMEASE TASK MANAGER® 23
EXECUTE AS STANDARD REPORT 23
EXECUTING SEVERAL SIAMEASE TASKS 23
TASK RUN TIME PARAMETERS 24
CREATE BATCH INPUT 25
LOAD TRANSACTION ONLINE 25
TYPICAL PROCESSING MODES 25
SWITCHING OFF AUTOMATIC BATCH INPUT SESSION 26
CHECKING DATA BEFORE LOAD 26
SOURCE 27
MANUAL PARAMETERS 28
OPTIONS AVAILABLE DURING PROCESSING 28
OPTIONS AVAILABLE AFTER PROCESSING 29
Chapter 6 DATA FORMAT 30
FORMAT OUTLINE 30
ANALYSIS TECHNIQUES 31
HEADER AND DETAIL LOGIC (Multiple records per transaction) 32
BOTTOM UP AND TOP DOWN ANALYSIS 33
TOP DOWN Example 33
BOTTOM UP Example 33
TABLE LOGIC ANALYSIS 34
TABLE LOGIC Example 34
Chapter 7 DATA MAINTENANCE 35
MAINTAINING BUSINESS DATA 35
DATA MAINTENANCE PROCEDURE 36
AUTOMATING ROUTINE FUNCTIONS 37
DATA MAINTENANCE CONSIDERATIONS 38
DATA SOURCE 38
IDENTIFYING DATA THAT HAS BEEN CHANGED 38
KEY AND SELECTION ARE DUPLICATED 39
Chapter 8 PREPARING LOAD DATA 40
PREPARING DATA IN A SPREADSHEET or DATAFILE 40
OPTIONALLY FILLING FIELDS 41
LOADING DATA USING EXISTING DATA 41
IMPORTING DATA FROM MEMORY 41
VARIANT DATA (ENHANCED SIAMEASE VARIANTS) 42
SYSTEM FIELDS ($SY-MSGV2) 42
PARAMETER ID's (%BUK) 42
FORMULA (=DAYS?7) 42
VARIANT FORMULA 43
Chapter 9 SIAMEASE TASK MANAGER ® 44
OVERVIEW 44
THE TASK MANAGER PROVIDES FUNCTIONALITY TO 45
TASK MANAGER ICONS AND SYMBOLS 45
TASK MANAGER FUNCTIONS 46
CREATING NEW TASKS 46
CHANGING EXISTING TASKS 46
EXECUTE TASK 46
TASK STATUS 46
TASK CONFIGURATION MENUS 46
SCRIPT FUNCTIONS 47
MAINTAINING TASK DETAILS 48
ADDING EXTERNAL PROGRAMS INTO TASK MANAGER 50
ATTACHING SAMPLE DATA TO TASK 51
ATTACHING DOCUMENTATION TO TASK 51
Chapter 10 INTERFACING 52
INTERFACE TYPES 52
INTERFACE TEAM DYNAMICS 52
INTEGRATION INTO AN INTERFACE 53
SAMPLE INTERFACE CODE 54

Chapter 1

FOREWORD

Special thanks to all the people who have helped with the development and testing of Siamease. Many specialist consultants have volunteered their business and technical advice to make Siamease a product that most SAP projects cannot be without.

Siamease provides a wide range of functionality. It does this by exploiting SAP automation, thus allowing users to execute SAP transactions in bulk using data that has either been uploaded from a legacy system or data that already existing within the customer SAP database. Users are therefore able to perform data migration, build interfaces, maintain live customer data, build training systems and build test scripts.

This document explains all options available within Siamease. Separate documents exist that walk through business scenarios. Siamease consists of 4 components, a task builder and manager and a script builder and manager.

Please feel free to email your comments to us at simmonds@global.co.za

Chapter 2

DOCUMENT SPECIFICATIONS

Prepared by:	Simmonds and Associates
Released:	2001-04-27
SAP Versions:	4.6
Audience:	SAP Implementation Teams, Super Users, Development team, Functional Consultants, Data Administrators, Training Teams, Basis stress testers
Trademarks	Microsoft Excel ® is a trademark of Microsoft.

Chapter 3

SETUP

- Import Transport Requests
- Register License
- Define Configuration

IMPORTING TRANSPORT REQUESTS

Refer to the installation documentation provided with the installation transport request.

- FTP transport requests into the /usr/sap/trans/data and /usr/sap/trans/cofiles directories.
- Add transport request to import queue.
- Import transport request using an unconditional mode.

REGISTRATION

A license can be obtained by providing a SAP installation number to Simmonds and Associates. A code will be issued, valid for a prescribed period of time. To enter the code, execute transaction ZEZE. From the environment menu select “Register Siamease” and enter the supplied code.

SIAMEASE CONFIGURATION

Siamease classifies all work done in Siamease into two dimensions, the first being the functional application and sub-application definition, the second being the activity or project. The Siamease transport preloads the system with a set of projects, applications and sub-applications.

TO CONFIGURE ADDITIONAL APPLICATIONS:

- Execute transaction SM31.
- Enter table T9CS2 to define language specific application codes.
- Enter table T9CS3 to define language specific sub-application codes.

Siamease Project Applications

L...	A...	Short text
EN	AM	Asset Management
EN	BC	Basis
EN	CF	Corporate Finance
EN	CO	Controlling
EN	FI	Finance
EN	HR	Human Resources
EN	MM	Materials Management
EN	PA	Profitability Analysis
EN	PM	Plant Maintenance
EN	SD	Sales and Distribution
EN	TR	Treasury
EN	WM	Warehouse Management
EN	XX	Other

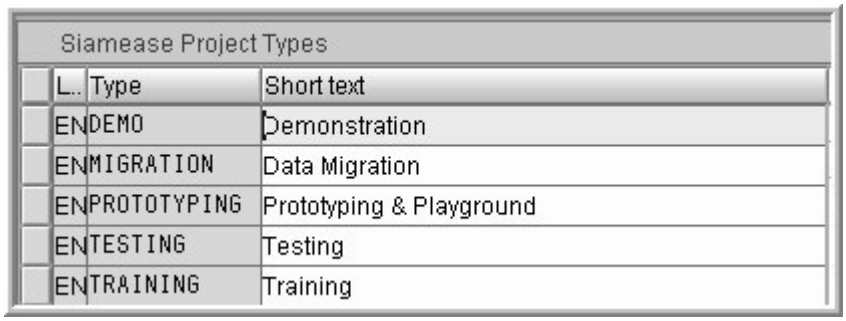
Siamease Project Sub Application

L...	Su...	Short text	A...
EN	AM	General	AM
EN	AMMA	Master Data	AM
EN	BC	General	BC
EN	FI	General	FI
EN	FIAC	Chart of Accounts	FI
EN	FIAR	Accounts Receivable	FI
EN	MM	General	MM
EN	MMMA	Master Data	MM
EN	MMMM	Material Movements	MM
EN	SBUP	Central Business Partner	TR
EN	SD	General	SD
EN	SDMA	Master Data	SD
EN	SDOR	Sales Order Processing	SD
EN	SDPR	Pricing	SD
EN	SDVF	Invoicing	SD
EN	SDVL	Deliveries	SD
EN	TRCA	Credit Risk Analyser	TR
EN	TRLO	Loans	TR

TO CONFIGURE ADDITIONAL PROJECT TYPES:

Siamease links projects to a project type. TEST is defined as a project type. All stress testing can be classified under the project STRESS test.

- Execute the Siamease Task builder (transaction ZEZE)
- Select the menu Environment > Project Types

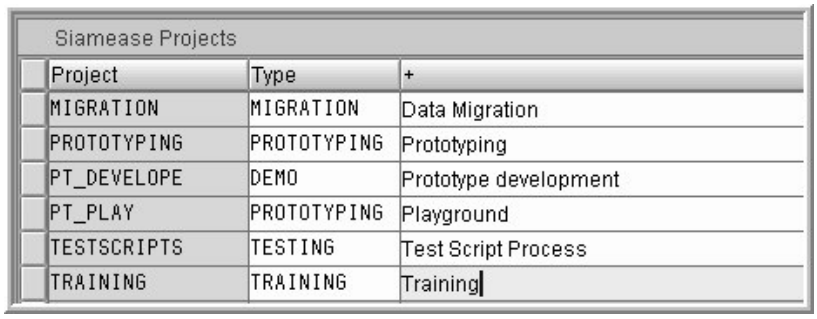


Siamease Project Types

L..	Type	Short text
EN	DEMO	Demonstration
EN	MIGRATION	Data Migration
EN	PROTOTYPING	Prototyping & Playground
EN	TESTING	Testing
EN	TRAINING	Training

TO CONFIGURE ADDITIONAL PROJECTS

- Execute the Siamease Task builder (transaction ZEZE)
- Select the menu Environment > Configure Projects



Siamease Projects

Project	Type	+
MIGRATION	MIGRATION	Data Migration
PROTOTYPING	PROTOTYPING	Prototyping
PT_DEVELOPE	DEMO	Prototype development
PT_PLAY	PROTOTYPING	Playground
TESTSCRIPTS	TESTING	Test Script Process
TRAINING	TRAINING	Training

NAMING CONVENTIONS

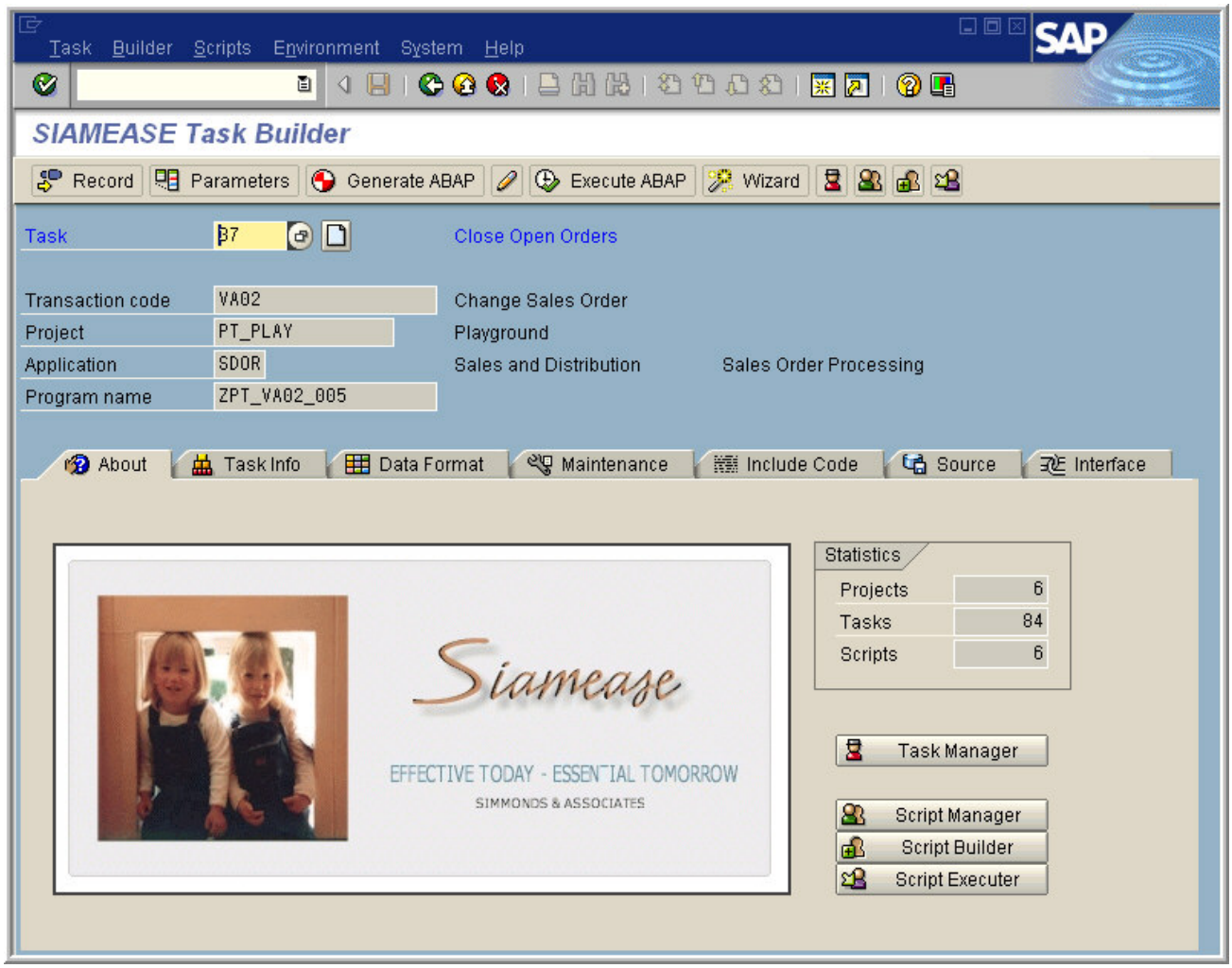
Siamease generates ABAP programs. The project name is derived from an application code, transaction code and a sequential number. Since the task manager manages all tasks, comments and program statuses can be maintained. Program names can therefore be named according to their function.

Chapter 4

SIAMEASE TASK BUILDER ®

TASK DEFINITION

A Task defines a specific business operation performed on a SAP transaction. Behind every task is an ABAP program capable of capturing data into the transaction. All tasks have status control allowing users to activate or deactivate a task. Documentation and other comments can be attached to a task and are accessible via the Siamease Task Manager ®. Other attachments include scenario data; spreadsheets, data files and Siamease enhanced run-time variants. Tasks can be chained together to perform a script – managed by the Siamease Script Manager ® and compiled using the easy Siamease Script Builder ®.



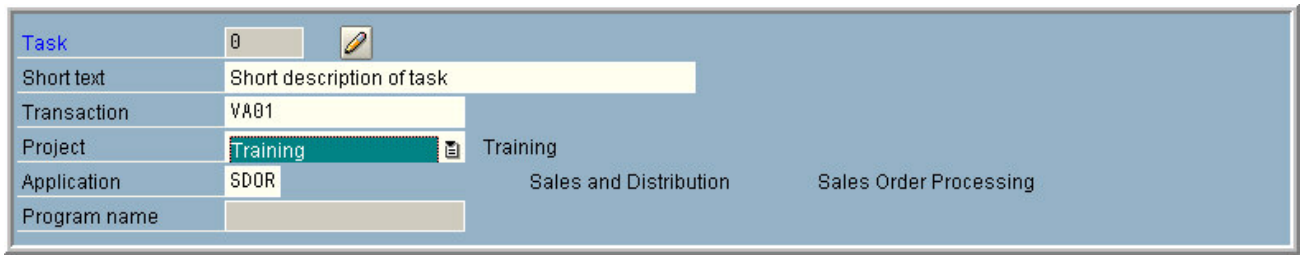
Although tasks are managed via the Siamease Task Manager® it is the Siamease Task Builder® that builds new tasks and creates template spreadsheets if need be.

The Task Builder is started, from within SAP, using transaction ZEZE.

TO CREATE A NEW TASK

- Press the “Create new task” icon on the Task Builder.
- Enter a short description for the new task
- Enter the desired SAP transaction code
- Select a project and Sub-application code
- Press Enter.

Siamease will then determine the next available program name for the task. It does this analysis by accessing existing ABAP programs in the SAP repository.



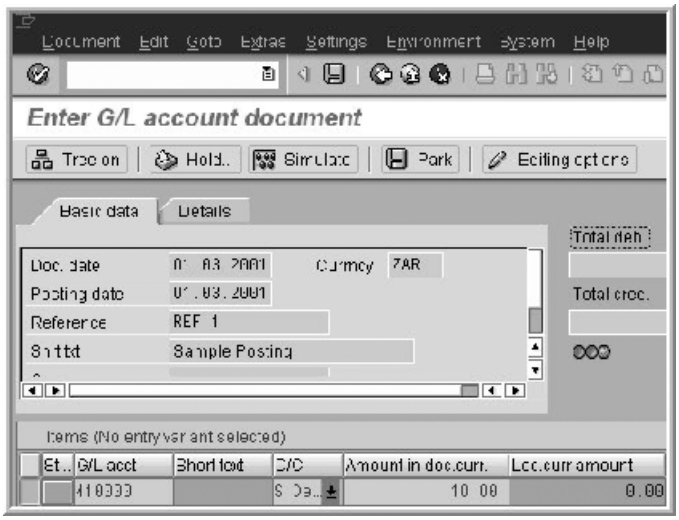
TO RECALL AN EXISTING TASK

- Enter the desired task number or search for a task using the value list or search help available. The search help available allows users to search for existing task using transaction codes, projects and applications.
- Press the change icon to switch from create mode to change mode

RECORDING THE TRANSACTION

Recording is used to train Siamease. It provides the opportunity for the functional user to demonstrate the business operation required using the SAP transaction.

- Press the record icon in the application toolbar.
- Capture the transaction as desired and save.



OTHER RECORDING CONSIDERATIONS

MANAGING THE PROBLEM

The trick to getting Siamease to generate the right code is in the way that the transaction is recorded. Often SAP transactions are designed to be user friendly but not program friendly. Through experience users identify ways of recording the transaction that makes the transaction easier to manage from a data migration perspective.

E.g. A SAP Business partner can contain the business partner details as well as numerous bank accounts. A user may create the business partner and all the bank account details in one transaction.
From a data migration perspective it may be easier to have a single pass that creates all the business partners with one bank account detail. Another simpler pass through the data can add multiple bank details to a business partner. Yet another pass can add additional addresses to the business partner, then other business partner relationships. In this way we break the problem down to a more manageable level.

SIAMEASE LOGIC

Adding logic to the transaction is what makes Siamease different. When users capture a complex transaction Siamease requests that a detail line item is repeated. Siamease identifies the repetition and from this identifies what fields belong to the header of the transaction, the detail and the footer.
If a transaction were recorded with two line items as suggested - other recording facilities would only be able to produce a data load that is only capable of loading 2 line items. Siamease detects trends and adjusts the ABAP code to allow for an unknown number of items to be captured. Provided that the capture technique demonstrated allowed for it.

RECORDING TIPS

- Note that mistakes are recorded. It is suggested that the user captures a trial transaction before recording to ensure that all values are known and that users do not have to hunt for codes during the recording. In essence the user should know how to capture the transaction before attempting to record.
- Match code / search help lookup are not usually recorded, but it is still advisable to perform these value look up functions on the session that you are not recording.
- Avoid using mouse clicks. Rather use menu options. Although not necessary, it sometimes helps to understand the transactions complexities. Remember that the task prepares a complete set of instructions to be sent to a transaction beforehand. It assumes that if a specific line is selected during the recording that that is the line to select always.
- Be careful of Defaults. Since a sample transaction has just captured certain fields may default. Always overwrite these with values, even if it is the same value (even typing one letter of the value will ensure that Siamease record the fields). The default may not appear later when the data is loaded.
- Avoiding unnecessary key strokes helps with the overall performance. Pressing Enter before pressing NEXT screen may be a safety net for users but not necessary for the task recording.

ADVANCED RECORDING

- Siamease by default uses an advanced recording option. This option allows the recording to detect the screens that will be called when the data is loaded (SAP on occasion will display one screen to a user but call a different screen when data is being loaded electronically. This causes problems for many other recording facilities.). To switch off the advanced recording select the DATA FORMAT tab and deselect the Advanced recording option.

RE-RECORDING A TASK

- A Task can be re-recorded in Change mode. It cannot be overwritten if the WRITE-PROTECT flag is set on the Task. The indicator can be maintained via the Task Manager and is visible on the TASK INFO tab.

SELECTING PARAMETERS (INPUT FIELDS)

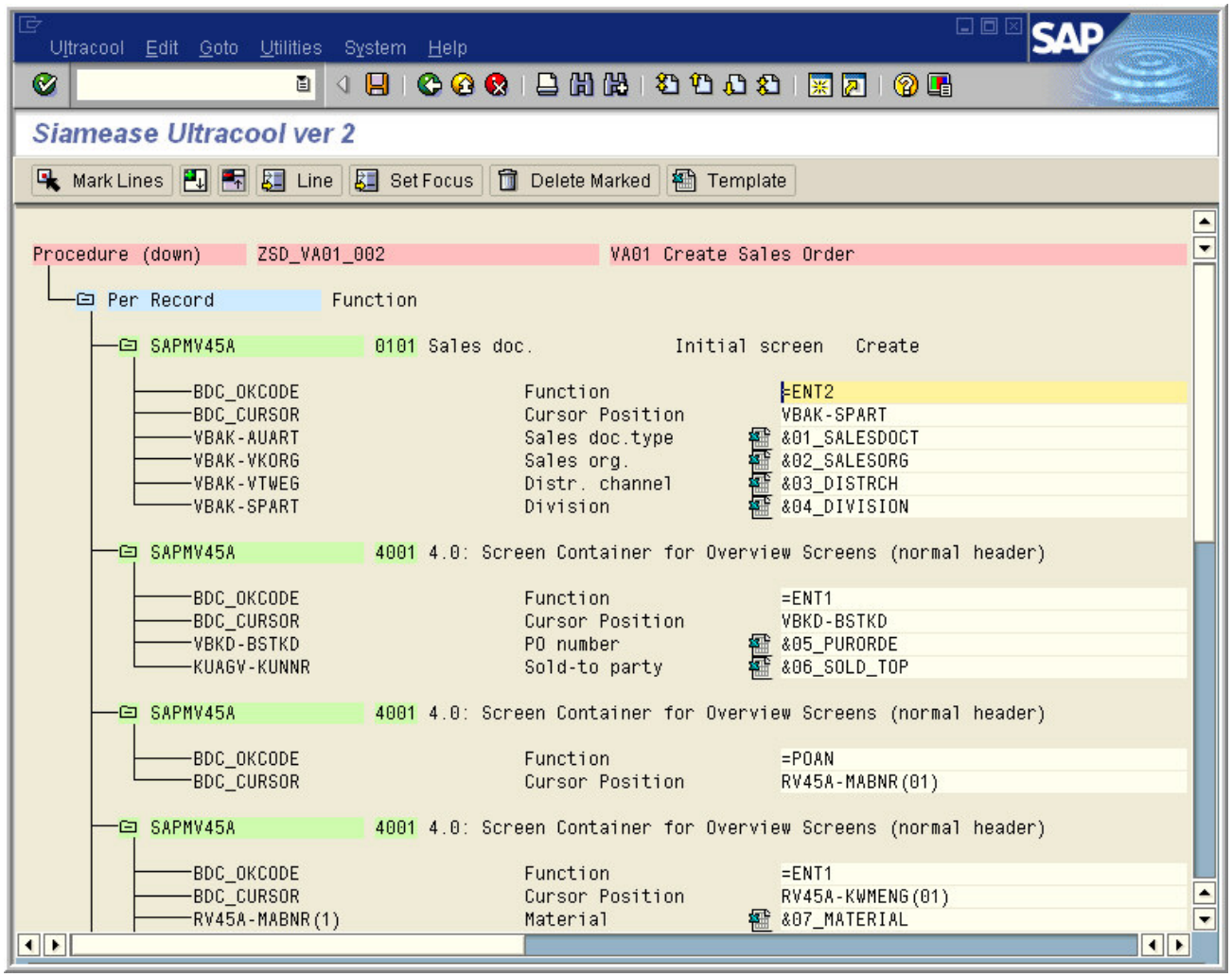
After recording a transaction every field captured has a hard coded value. In order to upload variable data, variable fields must be identified and flagged. Siamease can build a template spreadsheet for the variable data fields since these are the only fields that need data collection – all other non-variable fields will be loaded with constant values.

TO FLAG VARIABLE FIELDS

- Execute the Task Builder.
- Press the PARAMETERS icon on the application toolbar
- Siamease Ultracool is designed to display all the screens and fields that where captured during the recording in a hierarchical tree display. From here variable fields can be flagged, hard coded values can be checked, unwanted fields and screens can be deleted (advanced users only) and an Microsoft Excel ® spreadsheet can be generated.
- Expand the hierarchy list displayed until the desired fields are identified.
- Click on the icon next to each variable field.

The icon will toggle with each click. The default / hard-coded value will change to a unique parameter name. This name can be overwritten on the screen. Enter must be pressed after each line changed. Refer to Other Considerations regarding how Siamease determines the parameter name.

- Press the TEMPLATE button to create a spreadsheet template which can be populated with sample / scenario data and used to test the task.
- Press Save to save and return to the task builder.


Ultracool Edit Goto Utilities System Help
SAP
Siamease Ultracool ver 2
Mark Lines
Line
Set Focus
Delete Marked
Template
Procedure (down) ZSD_VA01_002 VA01 Create Sales Order
Per Record Function
SAPMV45A 0101 Sales doc. Initial screen Create
BDC_OKCODE Function =ENT2
BDC_CURSOR Cursor Position VBAK-SPART
VBAK-AUART Sales doc.type &01_SALESDOCT
VBAK-VKORG Sales org. &02_SALESORG
VBAK-VTWEG Distr. channel &03_DISTRCH
VBAK-SPART Division &04_DIVISION
SAPMV45A 4001 4.0: Screen Container for Overview Screens (normal header)
BDC_OKCODE Function =ENT1
BDC_CURSOR Cursor Position VBKD-BSTKD
VBKD-BSTKD PO number &05_PURORDE
KUAGV-KUNNR Sold-to party &06_SOLD_TOP
SAPMV45A 4001 4.0: Screen Container for Overview Screens (normal header)
BDC_OKCODE Function =POAN
BDC_CURSOR Cursor Position RV45A-MABNR(01)
SAPMV45A 4001 4.0: Screen Container for Overview Screens (normal header)
BDC_OKCODE Function =ENT1
BDC_CURSOR Cursor Position RV45A-KWMENG(01)
RV45A-MABNR(1) Material &07_MATERIAL

OTHER PARAMETER CONSIDERATIONS

FILLING MULTIPLE FIELDS WITH ONE VALUE

By default all fields are flagged as static values. Variable fields are those fields that will be loaded from an external source. Spreadsheets need only contain those fields that are not known. The others will automatically be captured with their constant value when the data is loaded. Parameter names can be copied and pasted from one field to another. If one date is filled into two fields then both fields on the screen can utilise the same parameter. In this case the spreadsheet data would only contain one date field / column.

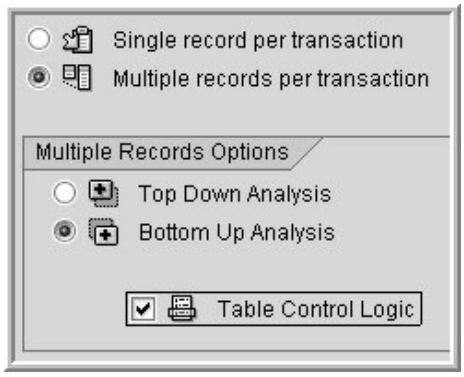
DATA FORMAT

Siamease has the ability to load multiple rows of data into one transaction. To achieve this the SAP transaction is recorded with a distinct header, 2 line items and footer. The Header data always prefixes the line item data and usually the footer consists of a simple SAVE. When loading data Siamease will analyse the header information against each line item. If the header changes it identifies the line item as a new transaction, closes the previous transaction (saves), captures the new transactions header and continues with the first line item.

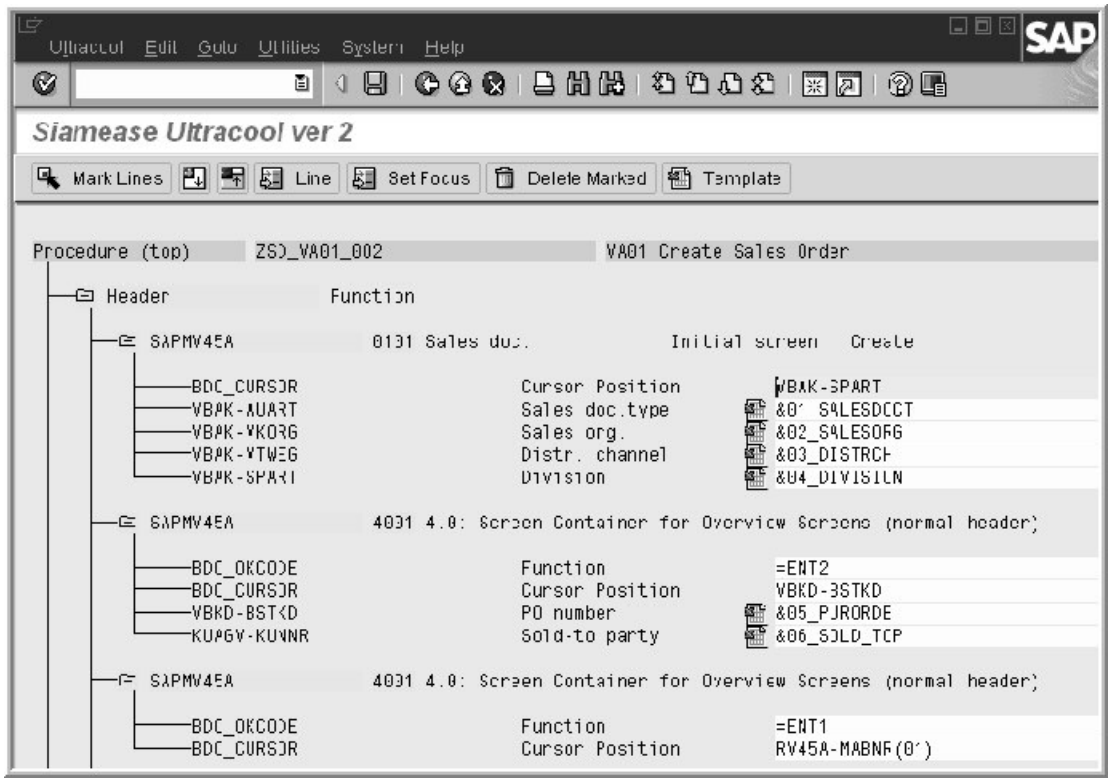
In this case the transaction recorded would have captured two line items and thus would have the same parameter screens and fields repeated in the recording. By default Siamease will assume that a simple transaction was recorded and will therefore indicate PER RECORD in the Parameter screen.

In this mode each parameter selected is deemed as being unique and thus gets a unique parameter number and name.

If the transaction indeed is to be used to load multiple rows of data to one transaction then it is advisable to flag the task as multiple via the DATA FORMAT tab.



In this mode Siamease will present the screens to the user in HEADER, DETAIL and FOOTER categories. When selecting variable fields, a check will first be done to see if the field already has been assigned a parameter name, if so it assigns the same parameter name. If this does not work, the user still has the option to overwrite the parameter name on the screen and press enter.



Refer to the section on loading multiple rows for more information.

DATA TEMPLATE

After selecting variable fields (before SAVING), press the TEMPLATE icon in the application toolbar. Siamease will open Excel and create a template spreadsheet. The contents of the template are primarily for documentation. When loading data, Siamease only requires the fields to be in the correct order and in a format compatible with the SAP capture screen (e.g. Dates in capture format).

	A	B	C	D	E	F	G	H
1	ZSD_VA01_002							
2	SalesDocTy	Sales org.	Distr. chl	Division	Pur. order	Sold-to pt	Material	Order qty
3	&01_SALESDOCT	&02_SALESORG	&03_DISTRCH	&04_DIVISION	&05_PURORDE	&06_SOLD_TOP	&07_MATERIAL	&08_ORDERQTY
4								
5								

It is suggested to save the template with a suitable name on your project network. This template can be filled with scenario data and later attached to the task. Refer to the Siamease Task Manager® for details on attaching data and documentation.

ADVANCED MAINTENANCE.

Often fields are left out when recording. The advances maintenance option allows these fields to be flagged and included after the recording. To maintain press F6 or select the Advanced Maintenance option on the Edit Menu.

GENERATING THE TASK

This option generates the ABAP program and creates a transport request for the task.

- Press the GENERATE icon on the application toolbar.

OTHER GENERATION CONSIDERATIONS

Siamease offers a wide variety of options when generating. The default settings suffice for the majority of the time. Most of the tabs on the Siamease Task builder relate to generation options.

TRANSPORTING

If the user is a registered developer on the system, Siamease will prompt for a development class and transport request. If not the ABAP program and task are still generated but will require a developer to transport at a later stage. The ABAP program can then be found under local objects for that user (SE80).

DATA FORMAT

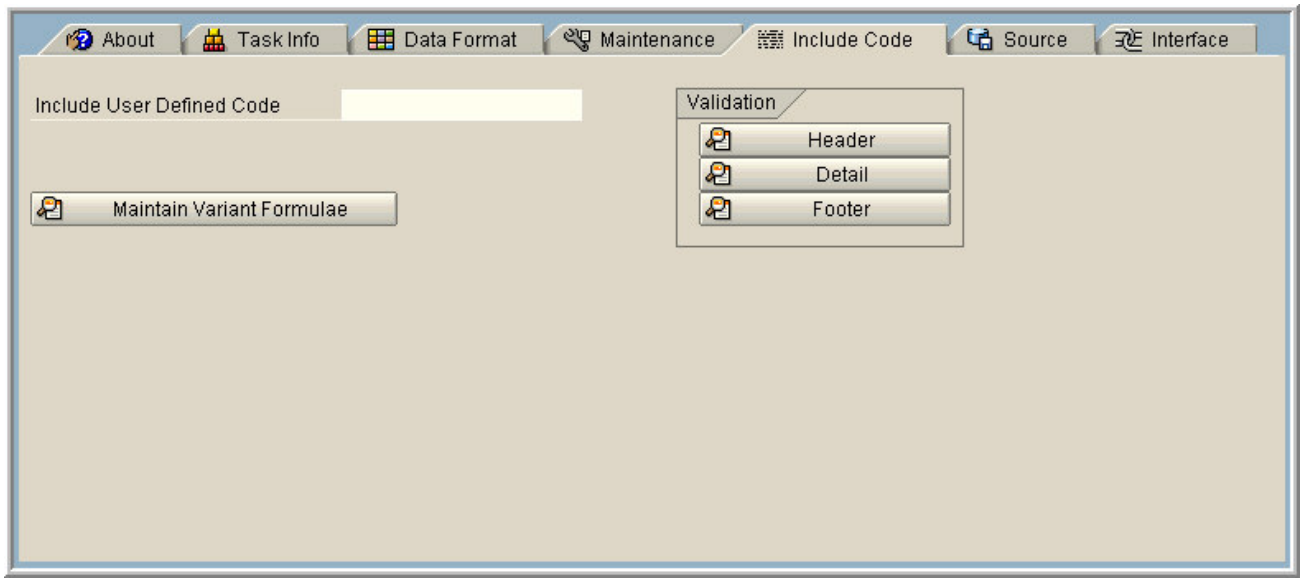
Refer to the DATA FORMAT Chapter.

DATA MAINTENANCE

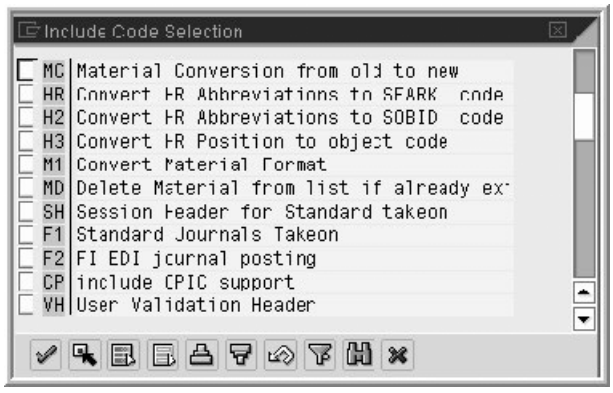
Refer to the DATA MAINTENANCE Chapter.

INCLUDING CUSTOMER CODE

Siamease allows additional code to be included into a Generated program. This feature allows developers to enhance certain components of a future generated program. Users can then choose to include the enhancements when they generate new tasks. Additional Code added to Siamease is identified by a 2 digit include code.



The INCLUDE tab provides a drop down list. Include codes can be strung together e.g. CU SY PI FM.

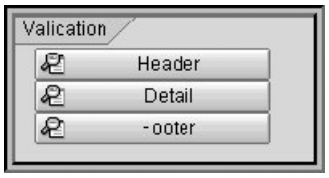

Include Code Selection
MC Material Conversion from old to new
HR Convert HR Abbreviations to SFARK code
H2 Convert HR Abbreviations to SOBID code
H3 Convert HR Position to object code
M1 Convert Material Format
MD Delete Material from list if already ex
SH Session Header for Standard takeon
F1 Standard Journals Takeon
F2 FI EDI journal posting
CP include CPIC support
VH User Validation Header

INCLUDE CURSOR POSITIONS

By default Siamease records cursor positions but does not include them into the generated programs unless told to. To include cursor positions, go to the INCLUDE tab on the main screen and enter CU as an include option (or use the search help / match code available). Cursor positions are only needed if the user actually selected an item from a report list.

GLOBAL VALIDATIONS

Global validations, such as authorizations, can be included by maintaining the header, items or footer routines - note that these must also be included via the include code identifiers - VH, VI and VF. (Refer to including customer code above.)



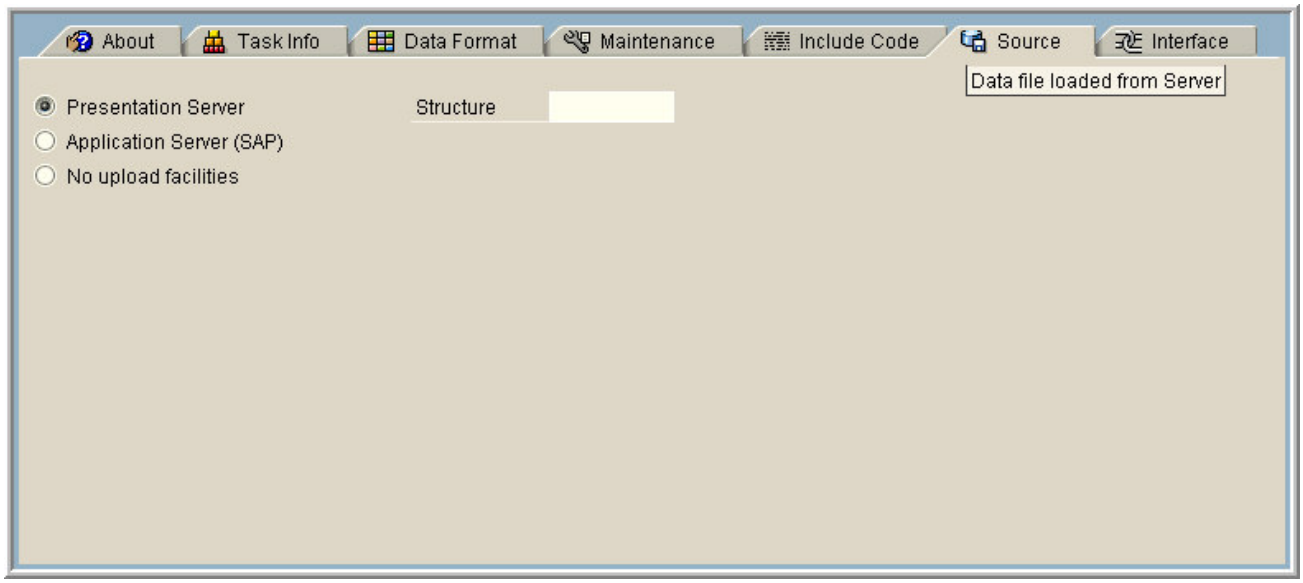
MAINTAIN VARIANT FORMULAE

Refer to the chapter on Preparing Load Data and the section on Variant Data.

DATA SOURCE

Data source allows the task to source its data file data from the SAP server. By default the data is uploaded from the users PC. On occasion a file could be copied onto the SAP server or created by another module within SAP. In this case the application server option must be set before generating the task. The structure field can be used if the record layout of the data file on the SAP server is defined within the Data Dictionary.

By default, Presentation server is used and should suffice for the majority of data loads. SAP, on the other hand, defaults to loading data from the SAP Server, which requires the user to create a text file and FTP the file to the SAP server before creating a Batch input session.



SIMPLIFYING DATA CAPTURE

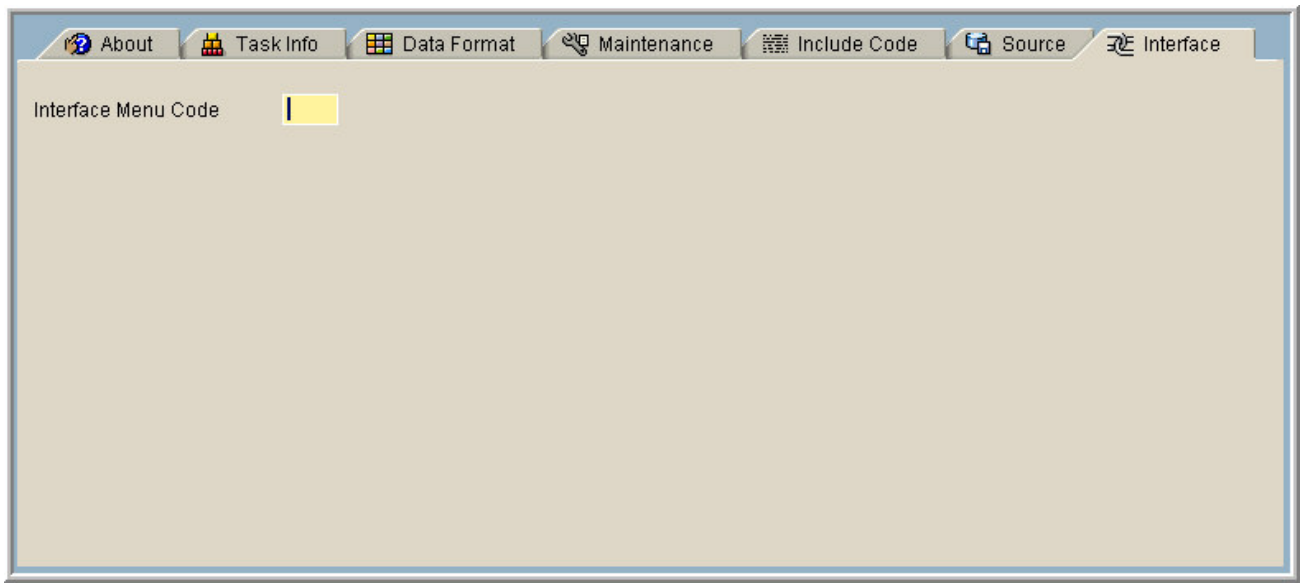
The “NO UPLOAD FACILITIES” check will generate a program that has a simple capture screen. The user would simply fill one screen of parameters and the Siamease program would capture the data into the complex transaction according to the business operation demonstrated by the functional user.

This option allows complex transactions to be simplified and used by external systems e.g. Internet via ITS or by bar-code scanner interfaces.

INTERFACING

This tab is used when the additional Siamease Co-interface is loaded. In this case a task can be added into the Co-interface manager by specifying a menu node for the task. This makes the task available to all applications using the Co-interface e.g. the intranet or bar code scanners.

Specify the co-interface menu code before generating the task, or simply regenerate the task at a later stage to add the task to the co-interface menu.



CO-INTERFACE NOTE

The co-interface module is a separate addition to Siamease. It is used to build interfaces on the fly that are accessible from systems outside of SAP. This provides immediate functionality to bar code scanners or intranet front-ends that pass data into SAP and receive data back. Although the server side of this software is ready, the client side is under development by a 3rd party.

Chapter 5

RUNNING A TASK

EXECUTING INDIVIDIAL SIAMEASE TASK

EXECUTE FROM THE SIAMEASE TASK BUILDER®

- Execute the Siamease task builder® via transaction ZEZE.
- Enter the relevant task number or search using the value request option.
- Press Enter to clarify the task details.
- Press the Execute icon on the application toolbar.

EXECUTE FROM THE SIAMEASE TASK MANAGER®

- Execute the Task manager via transaction ZETM
- Find the relevant task by expanding the task manager hierarchy tree nodes.
- Click on the task to be executed
- Press the Execute icon on the application toolbar.

EXECUTE AS STANDARD REPORT

- Follow menu path SYSTEM > SERVICES > REPORTING MENU
- Enter the ABAP program name into the selection screen
- Press Execute.
- This option is available in the live environment.

EXECUTING SEVERAL SIAMEASE TASKS

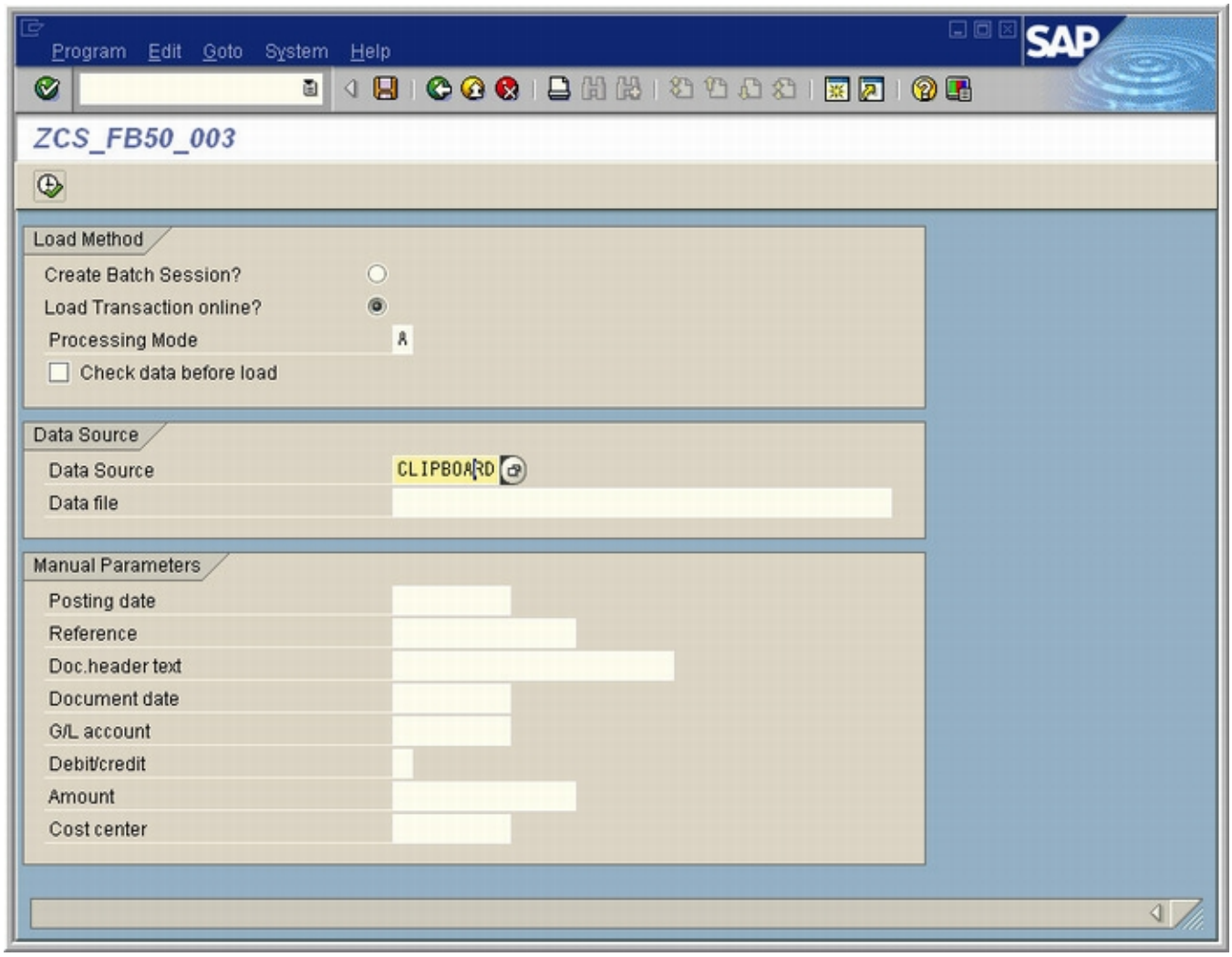
Refer to the chapters on the Siamease Script Builder ® and the Siamease Script Manager ®

TASK RUN TIME PARAMETERS

Each ABAP program generated follows specific standards and offers a variety of run time options.

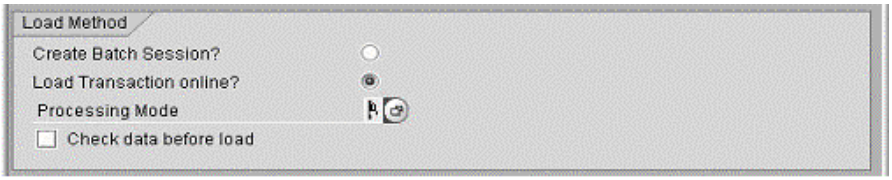
Each program has the following sections: -

- Load Mode - specifies how the program is to run.
- Data Source - specifies where the data source is.
- Manual Parameters - allows input data to be captured manually.



CREATE BATCH INPUT

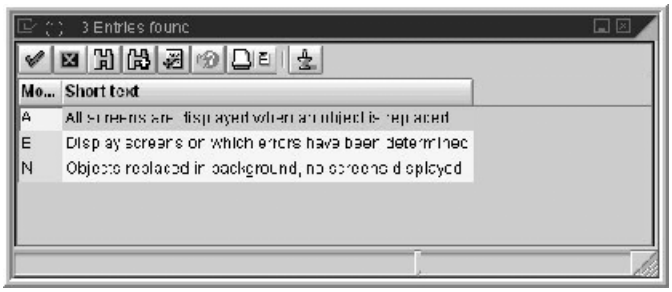
Siamease provides the ability to load data immediately or to store the data into a Batch input session where it can be processed at a later stage via transaction SM35.



LOAD TRANSACTION ONLINE

If the user chooses to load the transaction immediately (online) then a processing mode applies.
The following modes are available.

- **A** show all screens (testing)
- **E** only show screens with errors (default)
- **N** run in the background.



TYPICAL PROCESSING MODES

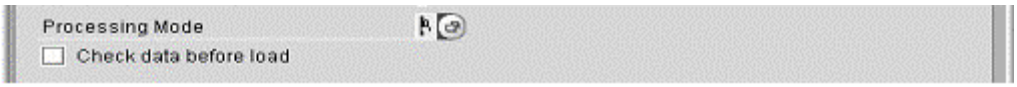
Typically mode A would be used to test the task. Mode E would be used when loading online. Mode N is typically used when interfacing since there is no user to correct any errors. In this case Siamease will store any failed transactions into a Batch input session that can be processed by a user via transaction SM35.

SWITCHING OFF AUTOMATIC BATCH INPUT SESSION

This option can be switched off if need be. The ABAP program has a hidden parameter that defaults to the users logon. A developer can easily change the program to default a blank name. In this case, Siamease will not create a batch input session. Note however that failed data would then be lost and would only be known by the Siamease post log / audit trail.

CHECKING DATA BEFORE LOAD

If the Check data before load checkbox is checked, an ALV report will be displayed before loading the data into SAP. This ALV report shows input / source data after validations and mapping have occurred. To continue processing the data, press the GREEN back arrow. To cancel the load press the RED cancel button on the standard menu. This report also provides functionality to email the load data, mail merge the data with MSWORD (on the fly) or create a Pivot table in Excel.



SOURCE

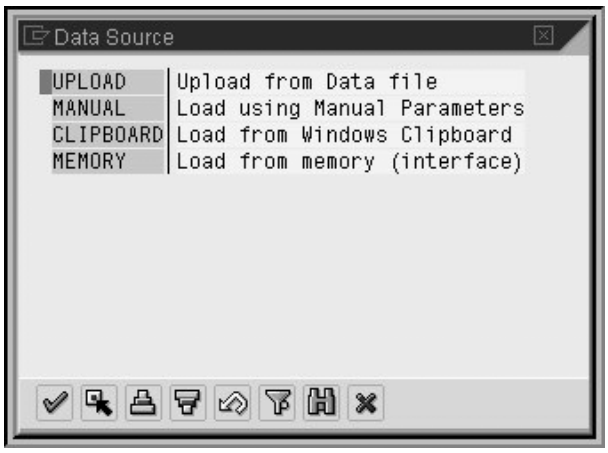
Source allows the Siamease task to identify at run-time where the data is being loaded.
The options available are: -

- **Data file**
  Create a text tab delimited data file from your spreadsheet. Do not include any headings. Specify the path and filename to the data file. Be sure to exit Excel first, since Excel locks the data file and prevents SAP from reading it.

- **Memory**
  Used to load data from an interface, where another ABAP program is collecting the data and passing the data to this program. See INTERFACING

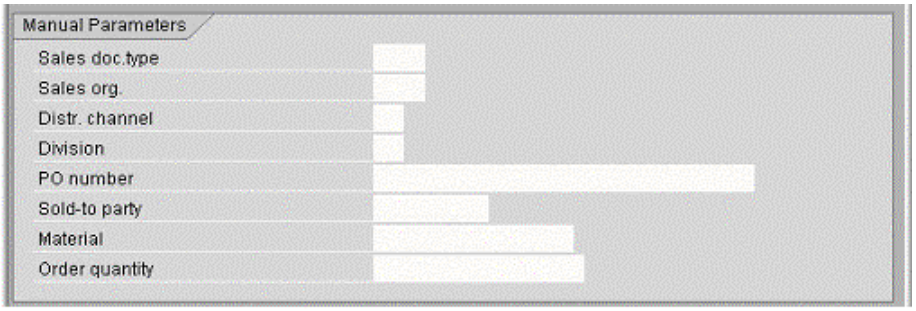
- **Manual**
  This option allows users to fill the manual parameters section (bottom half of task parameters), thus providing the users with a simple capture screen capable of loading complex SAP transactions. Refer to SIMPLIFICATION.

- **Clipboard**
  This option is the most common and is extremely useful. There is no data format preparation with this option. Simply highlight rows in a spreadsheet containing scenario data and press COPY. The Siamease task once executed will import the contents from the clipboard and process accordingly. This allows users to test a few rows of data before loading bulk. Do not include headings. This option allows data to be extracted from the middle of a spreadsheet.


Data Source
UPLOAD Upload from Data file
MANUAL Load using Manual Parameters
CLIPBOARD Load from Windows Clipboard
MEMORY Load from memory (interface)

MANUAL PARAMETERS

This option is typically used for data maintenance and simple data capture. If the Siamease task has been linked to a SAP table, then the selection criteria identified during the generation process are made visible here. This allows the user to filter the data that Siamease is using to perform the business operation.



Be sure to select MANUAL as the source of data.

OPTIONS AVAILABLE DURING PROCESSING

When the program is running, /N will cancel the current transaction being loaded and move to the next transaction. Run another transaction to get out altogether e.g. /Nzeze

OPTIONS AVAILABLE AFTER PROCESSING

If the “Check data before load” option is checked, Siamease will produce an ALV log / audit trail of the run. This log displays document numbers that have been created, system messages as well as the data that was uploaded. This log can be used for mapping. Press the Excel button to create a spreadsheet with all the relevant data.

Chapter 6

DATA FORMAT

FORMAT OUTLINE

There are two data formats that apply. The first is a simple format where 1 row of data processes 1 transaction. This would be used for a typical master data transaction. The other format allows for multiple rows of data to be processed in one transaction. In this case the data fields are grouped into header fields and line item fields. The header fields simply prefix each detail line. In other words the header fields are repeated for each line item. If a header fields has a different value to the line item above it, it assumes that this must be the start of a new transaction (since the header fields should remain static for each line item). This is typical for loading transaction data.

Simple format

Detail field one	Detail field two	Detail fields three	Detail field four	Detail field five
X1	Y1	Z1	123	456
X2	Y2	Z2	345	765

Header and Detail format

Header field 1	Header field 2	Detail field 1	Detail field 2	Detail field 3
Reference 1	01/01/2000	10	X1	123
Reference 1	01/01/2000	20	X2	456
Reference 1	01/01/2000	30	X3	789
Reference 1	01/01/2000	40	X4	012
Reference 2	01/01/2000	10	X1	123
Reference 2	01/01/2000	20	X2	456
Reference 2	01/01/2000	30	X3	789
Reference 2	01/01/2000	40	X4	012

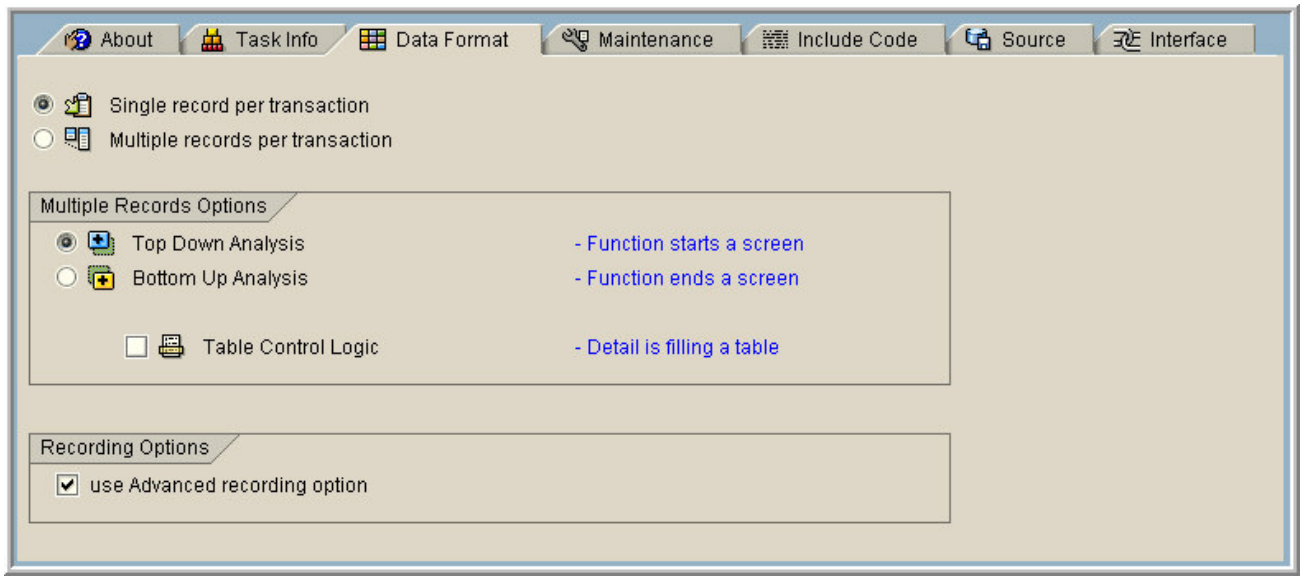
ANALYSIS TECHNIQUES

In order to determine which transaction fields are header and which are detail / line items, Siamease analyses the recorded transaction. Once it has determined a trend it can then determine that fields participating in the trend must be detail fields and other are either header or footer fields. Siamease provides 4 techniques for analyzing the recorded business operation.

It is this analysis that sets Siamease aside and is fundamental for the success of complex transactions. Very few transactions process a single row of data to a transaction. A financial journal, for example, will always require a debit and a credit to post correctly and more than likely will post an unknown number of line items. In the case where all data for a business operation is clearly defined and is completely known then a single record per transaction will suffice. If the business operation cannot clearly state how many line items would be posted into a transaction then the “Multiple records per transaction” option will solve the problem.

The possible options are available on the DATA FORMAT tab:

- Single record per transaction
- Multiple records per transaction – TOP down
- Multiple records per transaction – Bottom up
- Multiple records per transaction – Bottom up with Table control



HEADER AND DETAIL LOGIC (Multiple records per transaction)

In order for Siamease to correctly understand the business operation the user must capture the business operation or transaction in a way that Siamease can detect a trend. It is the trend that Siamease will replicate to capture the multiple rows of data. Each transaction must therefore also have a header and footer to complete the transaction. The header information must be static for the specific transaction else Siamease will capture a new transaction.

Siamease analyses the transaction and attempts to detect

- Header
- 1st line item captured and its subsequent screens
- 2nd line item captured and its subsequent screens.
- Footer – usually a SAVE and exit.

Not all transactions were developer to the same standards and thus the menu options available can vary. Siamease provides different analysis methods that can be applied after the transaction has been recorded. The Parameters screen uses these analysis methods to display the recorded screens to the user. If the results are incorrect then back out, change the data format options and retry the parameters screen. If no method works then the transaction should be re-recorded avoiding additional repetitions.

It is the repetition that Siamease detects. This indicates the 1st and 2nd line items should be captured the same way.

BOTTOM UP AND TOP DOWN ANALYSIS

In order to detect the 1st item and next item, Siamease looks for a single repetition. I.e. It will look to see if the next item was captured the same way that the first item was captured, by default it only looks at the screen and the function that was pressed on that screen (not the fields that were filled with exception to the TABLE LOGIC).

Some screens require a user to press a button in order to capture a value (e.g. INSERT NEW ITEMS), whilst others require a button to be pressed after values have been filled (e.g. PAGE DOWN). Siamease allows the user to choose different analysis techniques to match the transaction that was captured. This can be done on the FORMAT tab from the Siamease main screen.

TOP DOWN Example

- Assumes that the user captured a header
- Pressed [insert new items]. Captured an item.
- Pressed [insert new items]. Captured an item.
- Saved.

BOTTOM UP Example

- Assumes that the user captured a header and pressed ENTER.
- Captured an item. Pressed page down.
- Captured an item. Pressed page down.
- Saved.

TABLE LOGIC ANALYSIS

Table logic is used in conjunction with Bottom UP although it is a mixture of the two methods. It is used for Enjoy SAP postings and tables. Unfortunately it appears that the scroll bars in SAP do not record correctly and thus the transaction needs to be recorded to accommodate this. Table logic still requires a single repetition. The logic however assumes that the user identifies the first line on the table control and then the last line on the table control. The program that Siamease generates will capture a header, capture a full page of data from the first to the last line and then perform the function that was repeated during the recording (e.g. insert new items on the first line item - see below).

TABLE LOGIC Example

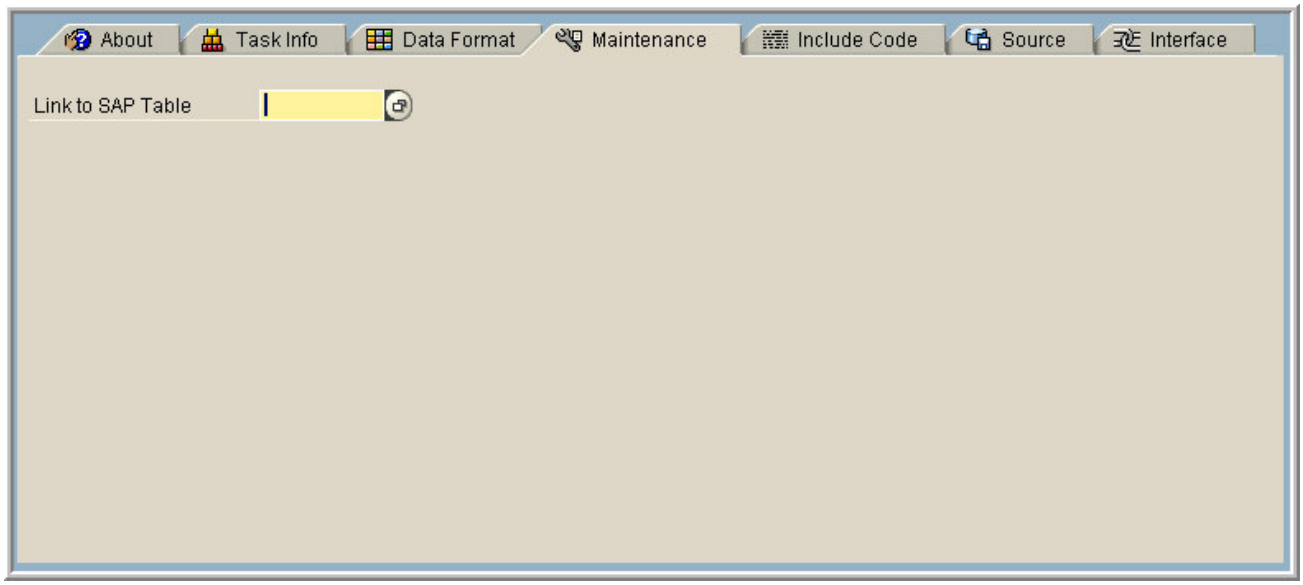
- Assumes that the user captured a header and pressed ENTER
- Captured the first line item on the screen. Selected the first line item and pressed [insert new items]
- Captured the last line item on the screen. Selected the first line item and pressed [insert new items]
- Saved

Chapter 7

DATA MAINTENANCE

MAINTAINING BUSINESS DATA

Data maintenance allows a business operation to be performed on a range of master data / transaction records existing in SAP. To setup a maintenance task, a change transaction is recorded. A SAP table is then accessed to provide the master data keys / values which are passed to the change transaction.



DATA MAINTENANCE PROCEDURE

Siamease assumes that the user wants to change a set of values from one value to a fixed known value (constant). E.g. the user wants to access a range of customer master records and set the sales statistic indicator on.

- Record a change transaction (as opposed to a create used during data migration). Be careful of default values on change transactions. Always overwrite all fields that you want to include in the business operation.

- Identify the key field as a variable. Typically the other fields would be hard-coded or will have a single parameter values

- Before generating, specify a SAP table that contains the relevant master or transactional data. Specify a SAP table on the Data Maintenance Tab

- Press the generate button

- Select fields from the specified table. A list containing all the fields in the specified table will be displayed. Select the field that matches the key to the business operation or transaction. Select other fields that will be used to filter the data to avoid processing ALL master records.

- Continue.

Siamease alters the standard generated program by replacing the manual parameters with the selection fields identified.

AUTOMATING ROUTINE FUNCTIONS

Besides changing master data, it is often a requirement to maintain transactional data. Often the function is quite simple but due to volumes within the transactional database, the task becomes resource intensive.

Example: a set of sales orders might require an availability check to be performed on a routine basis. Using Siamease perform an availability check on one sales order. Link the routine to the Sales order header table (VBAK) and flag the sales order number as one of the selection criteria fields. Dates and other indicators can be included to ensure that the correct set of Sales orders is included. This type of program could then be executed on a nightly basis via a job. Always check this in the test environment prior to going live since it has the potential to change data that you may not wish to be changed.

DATA MAINTENANCE CONSIDERATIONS

DATA SOURCE

This option should always use the MANUAL source of data. Always ensure that this is the case.

IDENTIFYING DATA THAT HAS BEEN CHANGED

It is advisable to check the CHECK DATA BEFORE LOAD checkbox. A report will be displayed prior the change, identifying the records that will be changed. Press the Green back arrow to continue processing, press the red cancel button to cancel the load. This list can be downloaded to a spreadsheet or document as an audit trail. SAP standard change documents can be used to identify before and after images of the data that has been changed. If need be, the change document tables can be accessed as the SAP table to reverse the change. Tables CDHDR and CDPOS contain the changed data.

KEY AND SELECTION ARE DUPLICATED

If the task is executed and both the transaction key and the selection field both exist on the screen as a duplicate then a possible mismatch has occurred.

Example

```
Material Number [   ]            <- key field to the change transaction
Material Number [   ] to [    ]  <- selection field identified as the key
```

To remedy this, edit the program as follows:

1) From the task builder, find the relevant task number and press the EDIT icon .
2) Find: the LOAD form; WHEN 'MANUAL', then the move-corresponding line.
   After the move-corresponding add a line that moves the table field to the screen parameter and remove the reference to the duplicated parameter.

- Add : MOVE <SAP table>-<field> TO UPTAB-<screen field> .
- Remove : UPTAB-<screen field> = <parameter> .

Sample.

```
Form Load_Data…
…
when 'MANUAL'.
…
select * from mara where …
  move-corresponding mara to uptab.
  Move mara-matnr to uptab-matn1.   " ← add this line of code
* uptab-matn1 = matn1.              " ← comment out this line
  …
Endselect.
```

3) Remove the additional PARAMETER from the parameters section e.g. parameter MATN1.
4) Save and activate the modified program.

Chapter 8

PREPARING LOAD DATA

PREPARING DATA IN A SPREADSHEET or DATAFILE

The data load preparation begins with the transaction record and the parameter screen. The parameter screen will identify all fields that will be uploaded. It will also provide a template spreadsheet that can be populated. Many of the fields downloaded into the Sample Template are not relevant for the load but merely provide statistical information to the user to assist with data collection.

Please note for spreadsheets the following should be addressed

- Date fields should be in the formal ddmmyyyy (8 characteristics)
- Value fields should be in the format nnnnnn.nn (no commas)
- All fields should be treated as text.
- All fields should be left justified.
- Codes entered into the data should match the SAP codes
- Any formatting is ignored
- The column order of the fields that are highlighted / copied to the template is imperative.
- Other fields can exist to the left and right of the data highlighted.
- Ensure that all codes are valid codes for SAP
- To create a data file from Excel®
    - Save data portion of a spreadsheet to a new sheet (no headings).
    - Save new sheet as a text tab delimited data file
    - Before uploading always exit from Excel®, since Excel® locks the data file.
    - Upload to SAP using UPLOAD source option.

OPTIONALLY FILLING FIELDS

By filling a field with the value of / (forward slash) the load program will skip the field in question and leave its value alone for that transaction. This is useful when maintaining data since we may optionally want to maintain one of two fields on a screen. With the skip command, Siamease will leave whatever value is on the screen alone. Without it the spreadsheet would have to know what value is already in the field since by leaving the field blank in the spreadsheet would blank the value already present in SAP.

LOADING DATA USING EXISTING DATA

- Refer to Data maintenance for more information.
- The success of this option is dependant on the fields flagged as variable and the fields available in the identified SAP table. If there is a mismatch a minor change to the generated ABAP may be required to ensure that the table field populates the screen field / variable.

IMPORTING DATA FROM MEMORY

- Refer to INTERFACES for more information.
- This option is only necessary if the imported data contains more than 1 row of data. Otherwise the interface program could submit the Siamease task with parameters set and source = ‘MANUAL’.

VARIANT DATA (ENHANCED SIAMEASE VARIANTS)

SAP Standard functionality allows variants to be saved. Siamease enhances this functionality by allowing users to store variables and formula instead of fixed values. Variant names can be maintained against either the master Siamease task or against a Siamease Script.

Siamease scripts are used to chain tasks or business operations together to create a business process or flow. In this case documents or master data created by a task may be used by another task to perform other business operations.
Variants usually store unit data. Siamease allows a variant to contain formulas and other variables.

The options available are: -

- **$**<Field name>
- **%**<Parameter id>
- **=**<Formula>

SYSTEM FIELDS ($SY-MSGV2)

$<Fieldname> has access to all system variables. Using the above example the field would be filled with the last message variable 2. If the last transaction issued a statement like ‘document 1234 has been created successfully’, sy-msgv2 would then contain 1234. Our variant would then be filled with the value 1234.

PARAMETER ID’s (%BUK)

Fills the field with the current company code stored in parameter id BUK. To identify a parameter id, go to a transaction that changes the desired master data fields. Select technical help on the key field. If a parameter id exists it will be displayed in the help popup window.

FORMULA (=DAYS?7)

Calls a formula that adds 7 days to the current date and fills the field.
Refer to VARIANT FORMULA below.

VARIANT FORMULA

Formulae are coded in program ZCSEZEFM. To access this program, execute the task builder (zeze), select the INCLUDE code tab and press the button.

The example below is a typical formula.

- Value field is used to pass a value to the formula and to send a value back to the screen field.
- Return code is used to identify errors. If the formula were to detect an error (i.e. master data already exists) then a return code of 8 would abort the load of that particular row of data.

```
*---------------------------------------------------------
* FORMULA NWEEK
* The purpose of this form is to return next weeks date.
*---------------------------------------------------------
form NWEEK changing valuefield returncode.
  data: nextweek like sy-datum.
  clear returncode.
  nextweek = sy-datum + 7.
  write nextweek to valuefield.
endform.
```

Please note that program ZCSEZEFM must be transported to the live environment after any changes.

ENHANCED VARIANT CONSIDERATIONS

INCLUDING ENHANCEMENTS TO BULK DATA

By default Enhanced variant functionality is always available to manual parameters, it is not however available to other data loads unless specifically requested using an INCLUDE Code.

- PI Parameter ID's
- SY System messages
- FM Formulae

This option is off by default due to the impact it has on processing large volumes of data.

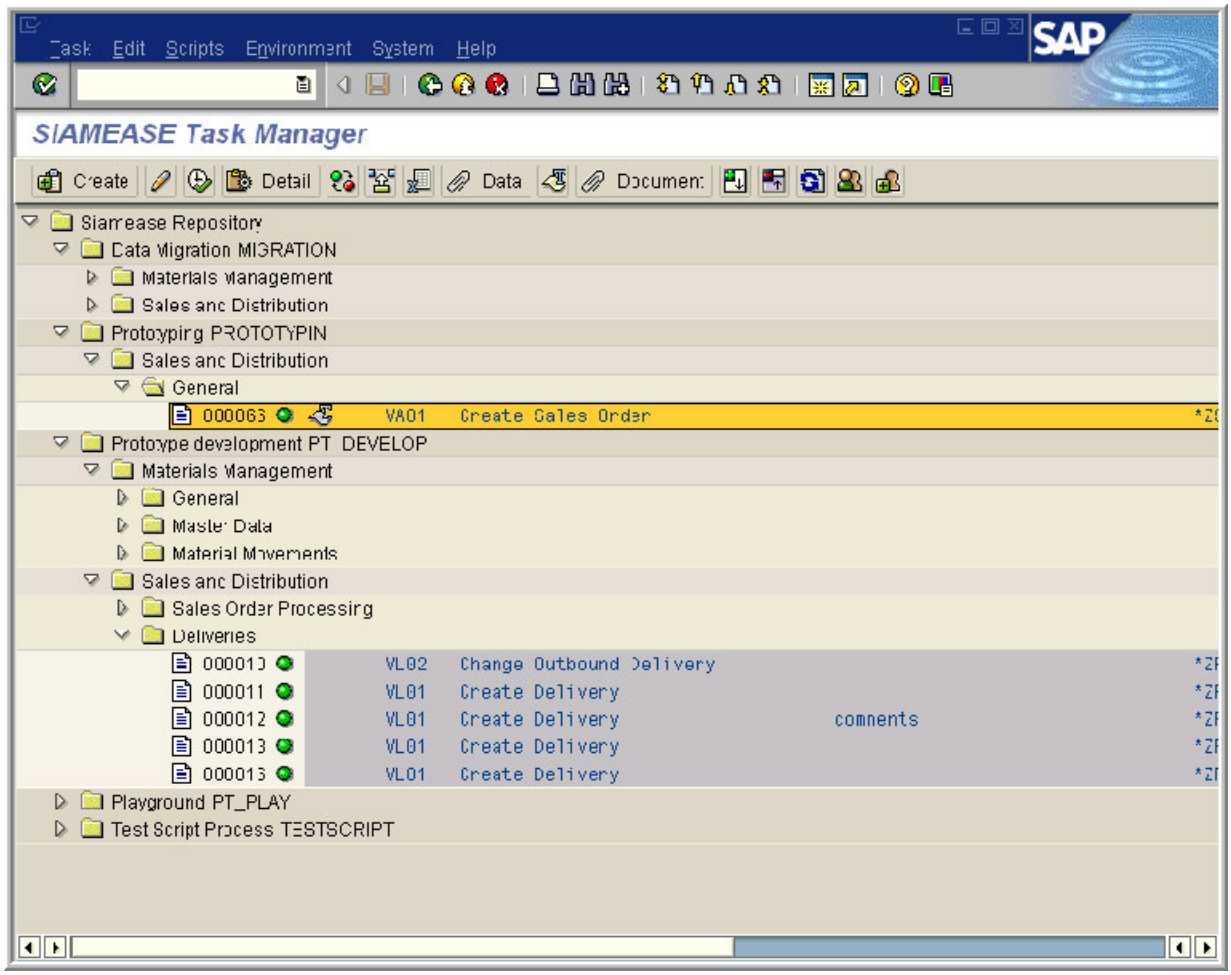
Chapter 9

SIAMEASE TASK MANAGER ®

OVERVIEW

The Task Manager provides a central control over task activities. From here tasks can be added and administered. New tasks that are created using the Siamease Task Builder ® are assigned to a Project and a Functional Application or area. The Task Manager provides a hierarchical display of the existing tasks, grouping by project and application. Common tasks are grouped together and access to tasks is easy.

To execute the Task manager press the icon (available on all Siamease screens) or execute transaction code ZETM.



THE TASK MANAGER PROVIDES FUNCTIONALITY TO

- Create new tasks (via the Siamease task builder®)
- Edit additional information / detail against each task
- Link attachments to a task
- View attachments for a task
- Execute a task
- Change task active – status
- Execute the script builder and script manager
- Administer tasks
- Link other non-Siamease programs into the task manager.
- Link to other Siamease configuration

TASK MANAGER ICONS AND SYMBOLS

	Each task is assigned a unique task number and is identifiable in the task manager with this symbol
	Active status of a task – red if inactive.
	Indicates that a documentation attachments exists for this task
	Indicates that a data attachment existing for this task.

TASK MANAGER FUNCTIONS

CREATING NEW TASKS

Press the button to execute the Siamease Task Builder® - refer to Task builder for more information.

CHANGING EXISTING TASKS

Select a task to be maintained by expanding the tree and single clicking on the task line.
Press the change detail icon in the application toolbar.
Siamease will execute the Task Builder and recall the task details.

EXECUTE TASK

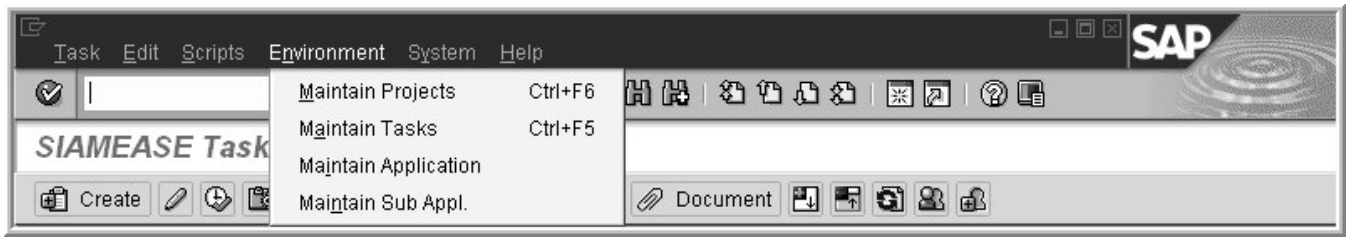
Select the desired task by expanding the application tree.
Single click on the task and press the EXECUTE icon.

TASK STATUS

Select task and press this icon to toggle task status between active and inactive. Only active tasks are available to script builder and script executor.

TASK CONFIGURATION MENUS

The following menu options allow Siamease Configuration to be amended.



SCRIPT FUNCTIONS

These icons access Siamease Script Manager® and Siamease Script Builder®. Script functionality in Siamease allows tasks to be linked / chained together to perform a business process flow. Stress testing, training and business process flow can be achieved by utilising existing tasks that have already been created using the task builder. Refer to the chapters on the Script Manager and Script Builder.

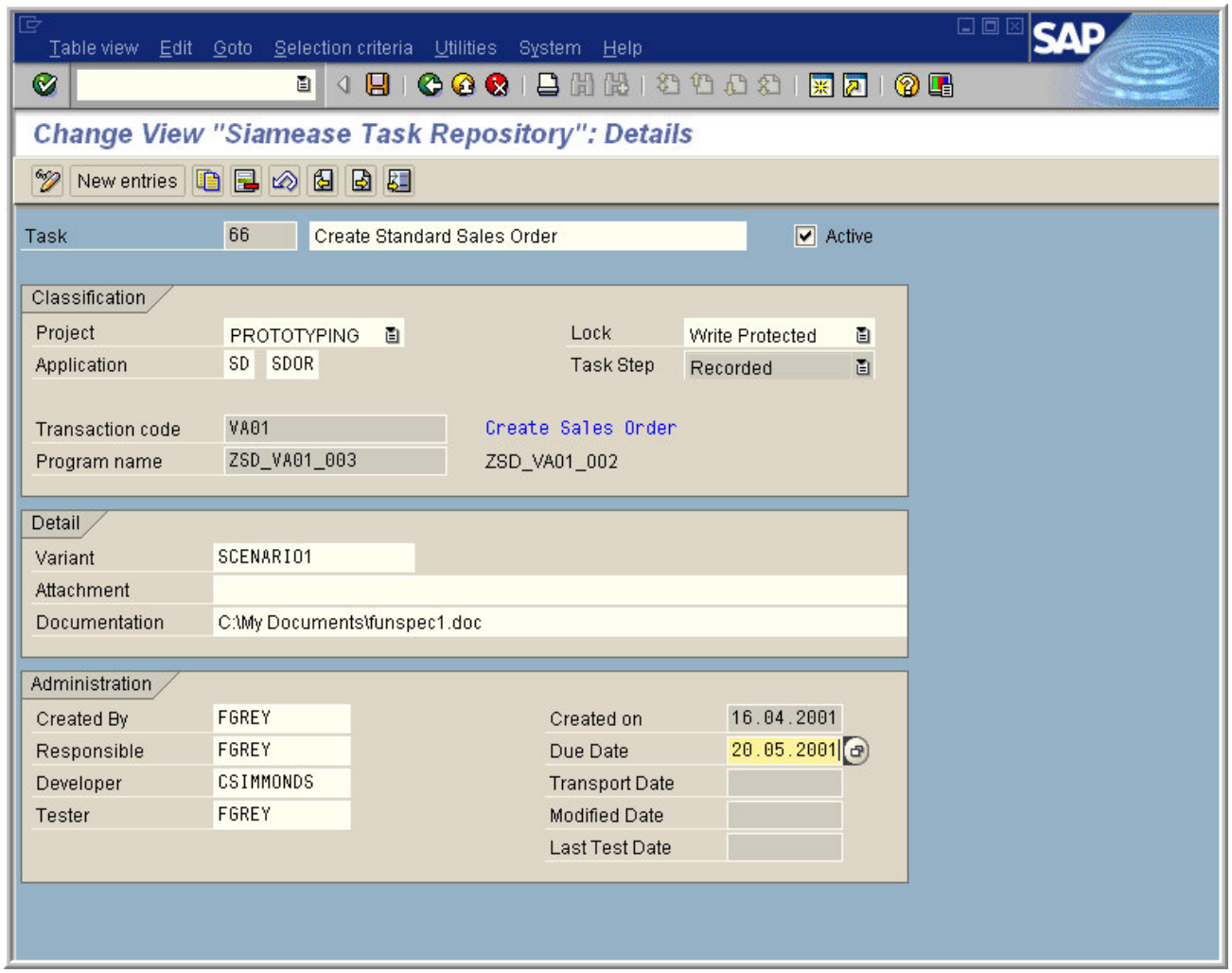
Or utilise the following menu paths.

MAINTAINING TASK DETAILS

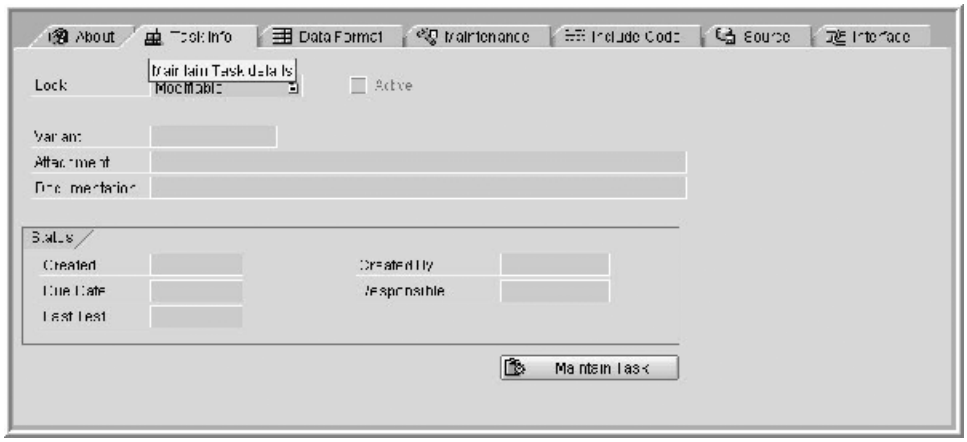
Select a task to be maintained by expanding the tree and single clicking on the task line.
Press the change detail icon on the application toolbar.

The task detail maintenance screen allows:

- Task description, application and project classification can be adjusted.
- The status of the task can be toggled.
- The Lock option allows the task to be write-protected to prevent the task from being accidentally overwritten.
- A default variant can be entered although this is an obsolete field.
- The filename of attachments (documentation and data) are accessible. If drive mappings change or the attachments are moved on the network then these file references can be adjusted.
- Users and dates can be maintained to keep track of responsibility.
- The Task Step identifies what step that the task builder is currently at regarding new task the task step has the following values. (Recorded, Parameters, generated, tested).
- After maintaining the task, press SAVE to return to the Task Manager.

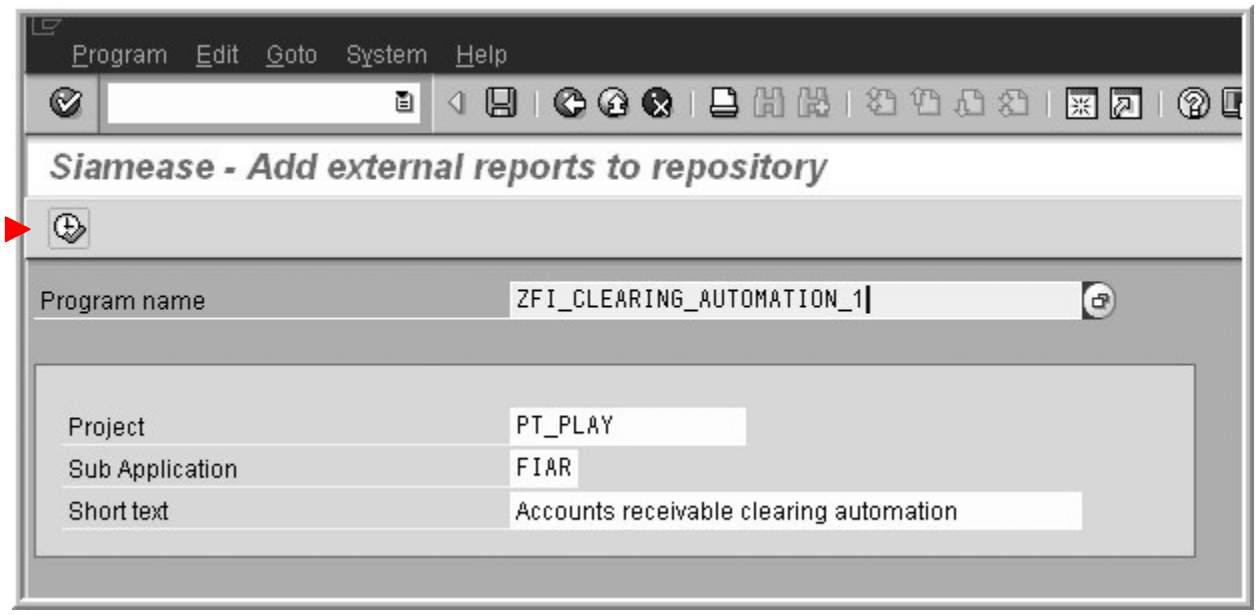


Task details are available on the TASK info tab in the Task builder. The Maintain Task button can perform the same function as the detail icon on the task manager.



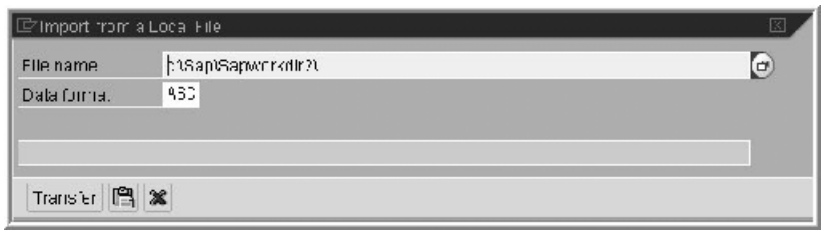
ADDING EXTERNAL PROGRAMS INTO TASK MANAGER

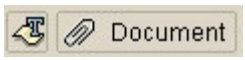
- Press the Add External Program Icon.
- Enter the external program name into the screen that pops up.
- Enter a Project code and Sub application code (application is derived from sub-application)
- Enter a short description of the task.
- Press the **execute** button to save.



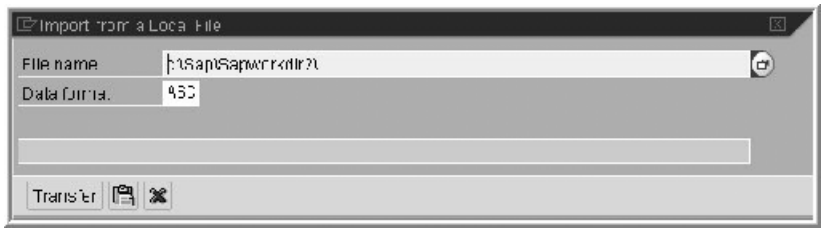
ATTACHING SAMPLE DATA TO TASK

- Press the paper clip icon to attach a sample data file.
  Browse network directory until desired attachment is found. Note that the attachment is merely a reference to the document. If the file is moved on the network, the new file location can be maintained against the task using the TASK DETAILS icon.
- Press the spreadsheet icon to open an attachment that has already been linked.



ATTACHING DOCUMENTATION TO TASK 

- Press the paper clip icon to attach documentation.
  Browse network directory until desired attachment is found. Note that the attachment is merely a reference to the document. If the file is moved on the network,
- The new file location can be maintained against the task using the TASK DETAILS icon. Press the word processing icon to open an attachment that has already been linked.



Attachments are indicated against the task as follows

Chapter 10

INTERFACING

INTERFACE TYPES

Interfacing is an option where another ABAP program will collect, prepare and pass bulk data to a Siamease generated program for posting into SAP R/3. Interfaces in general are split into two categories.

Temporary

- The interface will eventually be replaced when other SAP modules are implemented.
- The on-going interface is a user-operated interface.
- The user prepares the data, validates it and checks that it is being processed into SAP correctly.
- Cost of the development is a throwaway cost (therefore the cost of development should be low).

Permanent

- Automated interface that imports data begin the scenes.
- Programs instead of a user are doing validation and data mapping.
- Where the data for the interface is loaded into an intermediate interface area before processing.
- Interface performs an ongoing business function that justifies the extra expense of development.

INTERFACE TEAM DYNAMICS

Interfaces however usually utilise transactions in SAP that are more known to the business than to the technical development team building the interface. Siamease provides the means for both teams to work together to solve interface problems. The functional users use Siamease to create programs that can load data into their transactions. These programs are tested using variants or spreadsheets with sample data. This allows the technical team to manage the data flow, mapping and validation of data. The functional team can focus on the postings into SAP. If the functional team can define the Siamease programs first, it will provide a blueprint for the interface. Each component of the interface can be tested in a modular fashion independently to the other processes in the interface.

INTEGRATION INTO AN INTERFACE

Every program generated by Siamease automatically meets all temporary interface requirements. It also provides the ability for other interface programs to talk to it and pass bulk data to it. Each Siamease program defined has another SOURCE option called MEMORY.

This option works as follows

- The interface creates an internal table called UPTAB. UPTAB must be defined as it is in the Siamease generated program being called.
- The interface populates the internal table with data to be posted to SAP.
- The interface exports the internal table to memory
- The calling program submits the Siamease program.

SAMPLE INTERFACE CODE

In the Interface calling program define an internal table with the same structure as that defined in the Siamease generated program.

DATA:
begin of UPTAB occurs 0,
... field structure as defined ..
... in the Siamease program ..
end of uptab,
pfile(64).

PFILE = SY-UZEIT.

- Append data into internal table uptab.
- Export the bulk data to memory as follows

```
EXPORT uptab TO MEMORY ID pfile.
```

- Pfile is a unique memory identifier for this load.
- Call the Siamease program via a submit statement.

```
SUBMIT <ZSIAMEASE_PROGRAM>
    with source = memory
    with pfile = pfile
    with pmode = 'N'
    and return.
```

If the interface is only processing one record at a time then the calling program could simply submit the Siamease program and fill all the manual parameters using the submit WITH option and source = 'MANUAL'. No internal table would then be required.